**SQL Server Triggers**

SQL Server CREATE TRIGGER

The CREATE TRIGGER statement allows you to create a new trigger that is fired automatically whenever an event such as [INSERT](https://www.sqlservertutorial.net/sql-server-basics/sql-server-insert/), [DELETE](https://www.sqlservertutorial.net/sql-server-basics/sql-server-delete/), or [UPDATE](https://www.sqlservertutorial.net/sql-server-basics/sql-server-update/) occurs against a table.

1. Create a table for logging the changes

CREATE TABLE production.product\_audits(

change\_id INT IDENTITY PRIMARY KEY,

product\_id INT NOT NULL,

product\_name VARCHAR(255) NOT NULL,

brand\_id INT NOT NULL,

category\_id INT NOT NULL,

model\_year SMALLINT NOT NULL,

list\_price DEC(10,2) NOT NULL,

updated\_at DATETIME NOT NULL,

operation CHAR(3) NOT NULL,

CHECK(operation = 'INS' or operation='DEL')

);

2) Creating an after DML trigger

CREATE TRIGGER production.trg\_product\_audit

ON production.products

AFTER INSERT, DELETE

AS

BEGIN

SET NOCOUNT ON;

INSERT INTO production.product\_audits(

product\_id, product\_name, brand\_id, category\_id, model\_year,

list\_price, updated\_at, operation

)

SELECT

i.product\_id, product\_name, brand\_id, category\_id,

model\_year, i.list\_price, GETDATE(),'INS'

FROM

inserted i

UNION ALL

SELECT

d.product\_id, product\_name, brand\_id, category\_id, model\_year,

d.list\_price, GETDATE(),'DEL'

FROM

deleted d;

END

3)Testing the trigger

INSERT INTO production.products(

product\_name, brand\_id category\_id model\_year, list\_price)

VALUES (

'Test product', 1, 1, 2018, 599 );

SELECT \*

FROM production.product\_audits;

DELETE FROM production.products

WHERE product\_id = 322;

the trigger was fired and inserted the deleted row into

SELECT

\*

FROM

production.product\_audits;